

Sistemas UML

Aula 4

Diagrama de Sequência

Prof. Ms. José Antonio Gallo Junior

Objetivo da Aula

- Compreender a função do **Diagrama de Sequência** na UML.
- Identificar **atores, objetos e interações**.
- Interpretar e criar modelos de alto nível de funcionalidades do sistema.

Introdução:

O que é um Diagrama de Sequência?

O diagrama de sequência é uma solução dinâmica de modelagem em UML bastante usada porque incide especificamente sobre **linhas da vida**, ou **os processos e objetos** que vivem simultaneamente, e as **mensagens trocadas** entre eles para desempenhar uma função antes do término da linha da vida.

Um diagrama de sequência é uma espécie de **diagrama de interação**, pois descreve como, e em qual ordem, um grupo de objetos trabalha em conjunto. Estes diagramas são usados por desenvolvedores de software e profissionais de negócios para entender as necessidades de um novo sistema ou para documentar um processo existente.

Diagramas de sequência são conhecidos como **diagramas de eventos** ou **cenários de eventos**.

Introdução:

O que é um Diagrama de Sequência?

O **Diagrama de Sequência (Sequence Diagram)** é um **diagrama comportamental** da **UML** que mostra **como os objetos interagem ao longo do tempo**.

Objetivos:

- Representar **fluxos de interação** entre **atores** e **objetos**.
- Mostrar **ordem e tipo de mensagens trocadas**.
- Ideal para **visualizar um caso de uso em detalhes**.

Diagrama de Sequência

Os pontos positivos do Diagrama

Diagramas de sequência podem ser referências úteis para empresas e outras organizações. Experimente desenhar um diagrama de sequência para:

- Representar os detalhes de um caso de uso UML.
- Modelar a lógica de um processo, função ou operação sofisticado.
- Ver como objetos e componentes interagem uns com os outros para concluir um processo.
- Planejar e compreender a funcionalidade detalhada de um cenário existente ou futuro.

Diagrama de Sequência

Os pontos negativos do Diagrama

- **Complexidade:** casos de uso muito grandes ou complexos tender a ficar muito grandes para detalhar todos os pontos de execução.
- **Manutenção:** A medida que o sistema evolui, e for necessário alterar diagramas complexos a manutenção desses diagramas torna-se extremamente difícil e complicada.
- **Detalhar Visão de Alto Nível:** Não é ideal para detalhar uma arquitetura como um todo, mas partes dessa arquitetura, devido a complexidade e manutenção

Diagrama de Sequência

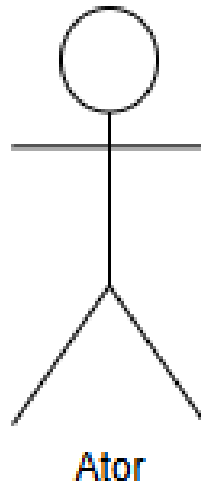
Cenários ideias para aplicação

- **Cenário de uso:** um cenário de uso é um diagrama de como seu sistema poderia ser utilizado. É ótimo para certificar-se de que você compreendeu a lógica de cada cenário de uso do sistema.
- **Lógica do método:** assim como você usaria um diagrama de sequência UML para explorar a lógica de um caso de uso, você pode usá-lo para explorar a lógica de qualquer função, procedimento ou processo complexo.
- **Lógica de serviço:** se você acredita que serviços são métodos de alto nível usados por clientes, um diagrama de sequência é ideal para mapeá-lo.

Elementos básicos do Diagrama de Sequência

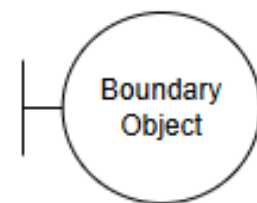
Atores

- Iniciam o fluxo.
- Representados com **bonequinho à esquerda** (ex: usuário, sistema externo).

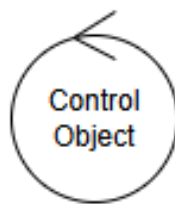


Elementos básicos do Diagrama de Sequência Objetos

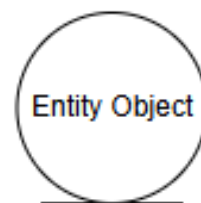
- Entidades do sistema que trocam mensagens.
- Representados como **retângulos com sublinhado** (ex: :Usuario, :SistemaLogin).



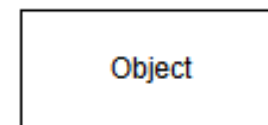
Representa
Interfaces Gráficas
API ou outros sistemas



Representa
Controladores,
Núcleo do Sistema



Representa
Entidades do
Banco de Dados

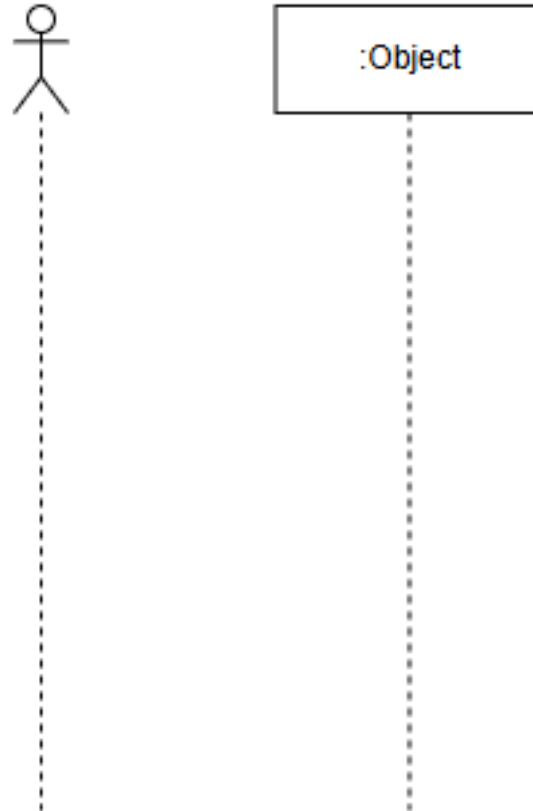


Representa
Qualquer OBJETO
do Sistema ou
outras Ferramentas

Elementos básicos do Diagrama de Sequência

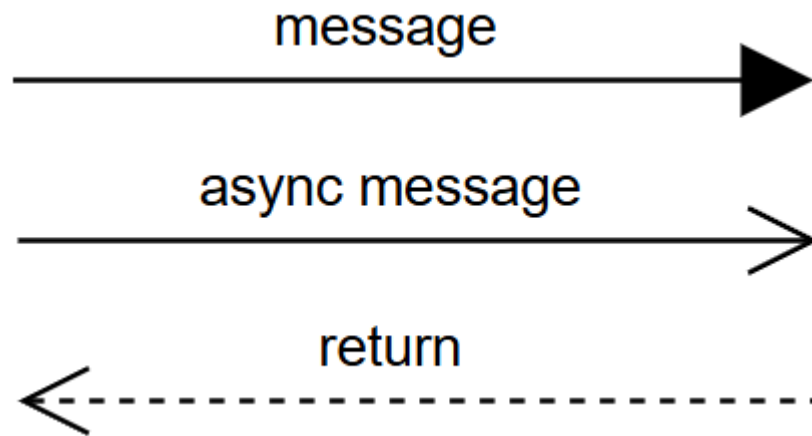
Linha de vida

- **Linha vertical** que representa o tempo de vida do objeto na interação.



Elementos básicos do Diagrama de Sequência Mensagens

- **Síncronas** (chamada direta, espera resposta): linha contínua com seta cheia.
- **Assíncronas** (envio sem esperar resposta): linha contínua com seta aberta.
- **Retorno** (responder uma requisição): linha tracejada com seta aberta.



Elementos básicos do Diagrama de Sequência

Caixa de ativação

- **Barra vertical estreita** sobre a linha de vida.
- Representa o **tempo** em que o objeto **está processando**.

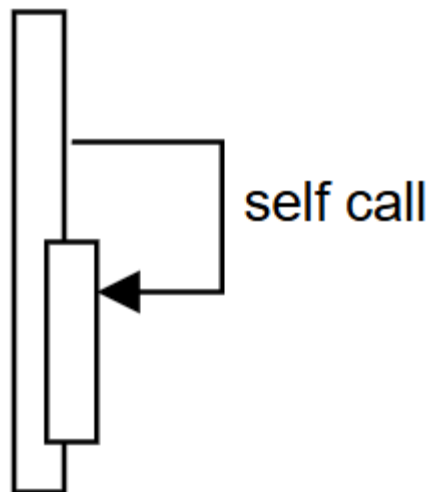


Diagrama de Sequência

Exemplo – Processo de Login

Cenário: Usuário acessa um site e tenta fazer login com usuário e senha.

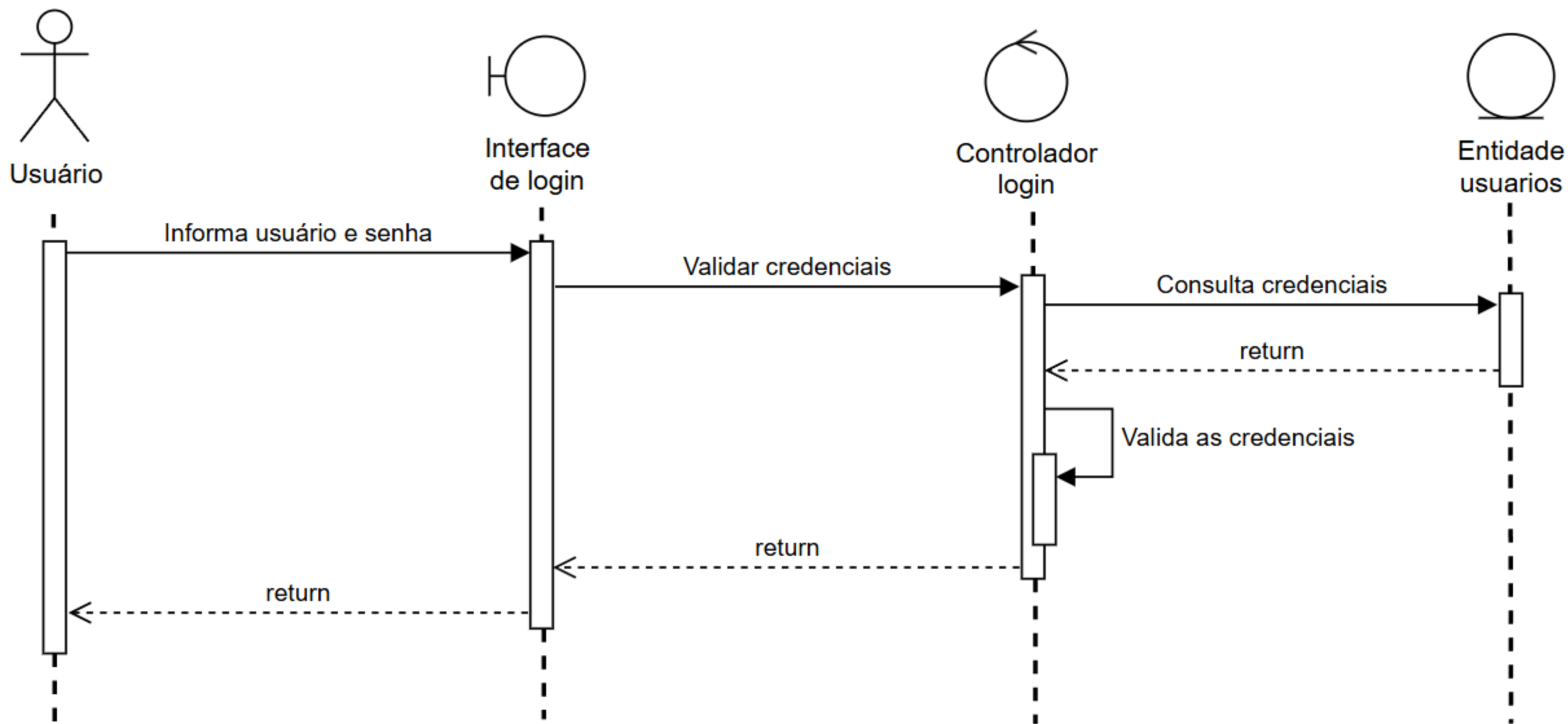
Devemos identificar:

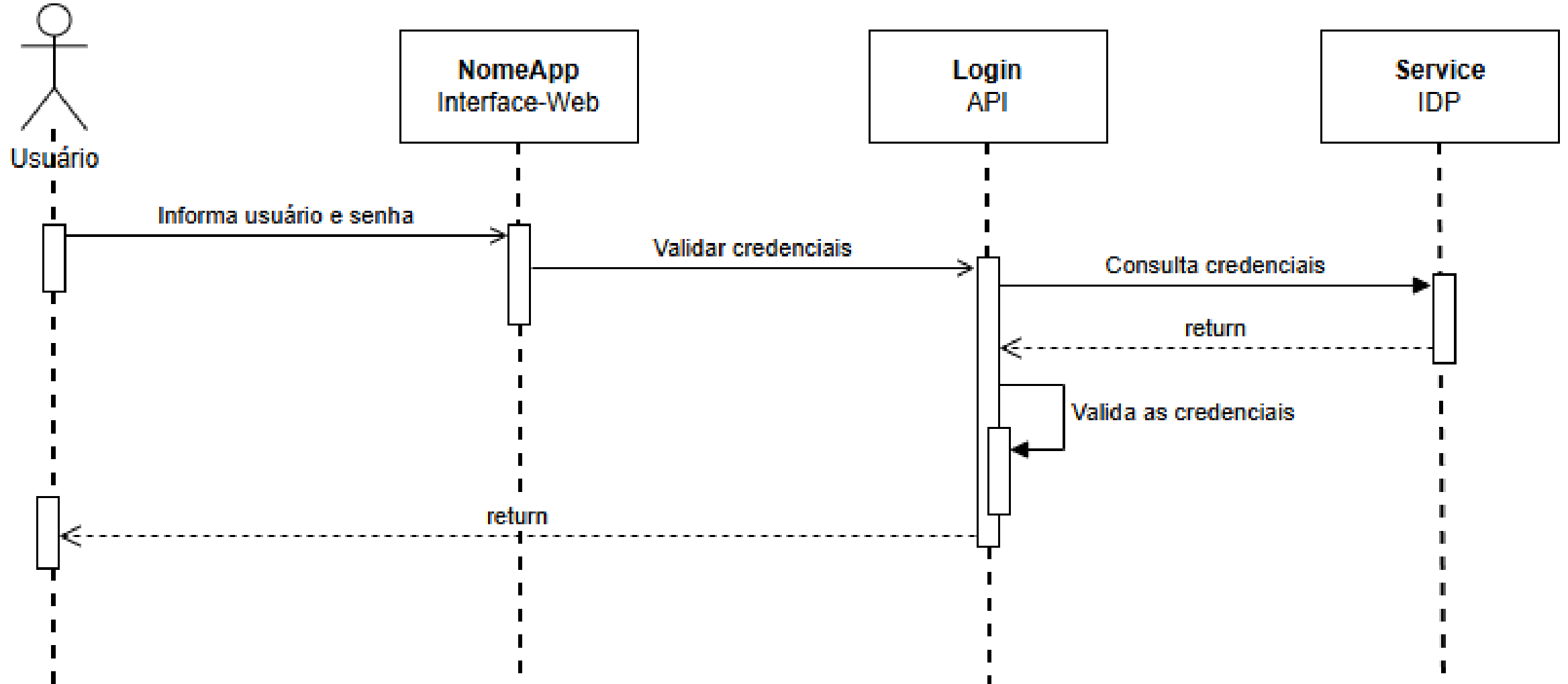
- Quem inicia a interação?
- Quais são as mensagens e o tipo?
- Onde estão as caixas de ativação?

Atores: Usuário

Objetos: Interface de login, Controlador login, Entidade de usuários

Mensagens: Informa usuário e senha, Valida credenciais e Consulta credenciais.





Atividade prática

Modelagem do processo de registro de usuário

- Modelar o processo de registro de novos usuários:
 - Atores: Usuário
 - Objetos: InterfaceRegistro, ControladorRegistro, BancoDeDados
- Devem identificar:
 - Quem inicia a interação?
 - Quais são as mensagens e o tipo?
 - Onde estão as caixas de ativação?

Diagrama de Sequência: Centralizados vs Distribuídos

Diagrama de Sequência

Diagrama Centralizado

- Um **único objeto controla o fluxo**.
- Os outros objetos apenas obedecem ou respondem.
- Mais simples, mas pode virar um “Deus do sistema”.

Exemplo: Um objeto **ControladorPedido** que faz tudo: valida cliente, acessa produto, cria pedido.

Diagrama de Sequência

Diagrama Distribuído

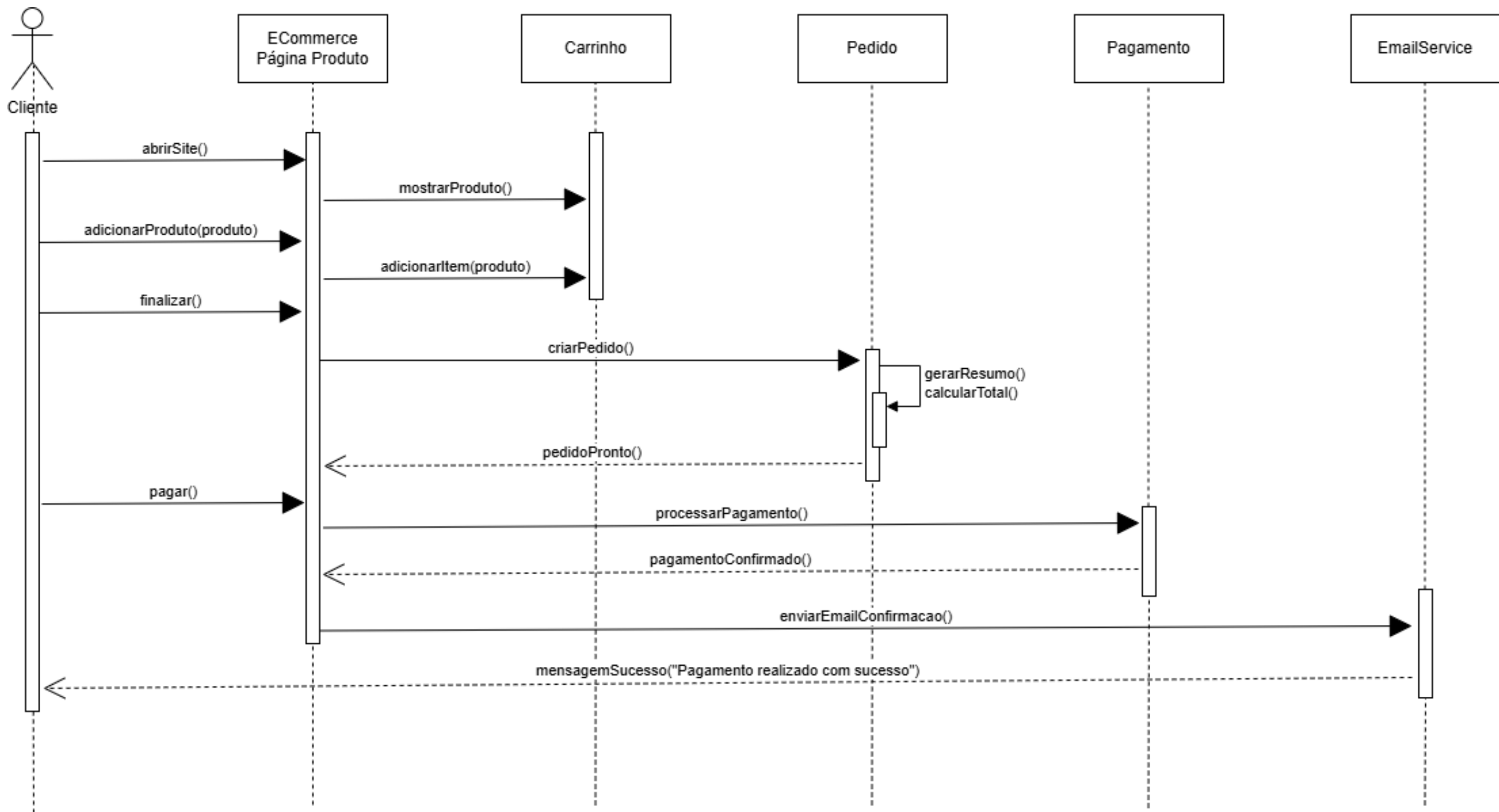
- **Responsabilidades são divididas** entre os objetos.
- Cada objeto executa sua própria parte da lógica.
- Melhor para **aderência à orientação a objetos** e **boas práticas**.

Exemplo: Cliente, Carrinho, Pedido e Pagamento colaborando entre si.

Exemplo prático

Compra no e-commerce

- Fluxo:
 1. Cliente acessa site.
 2. Adiciona produto ao carrinho.
 3. Fecha pedido.
 4. Realiza pagamento.
 5. Recebe confirmação.
- Objetos:
 - Cliente, TelaProduto, Carrinho, Pedido, Pagamento, EmailService



Atividade Prática – Diagrama de Sequência: Empréstimo e Renovação de Livro

A **biblioteca universitária** deseja um sistema que permita aos alunos realizar empréstimo de cópias de livros e, se necessário, solicitar a renovação da data de devolução.

Regras:

- O **aluno** deve estar **logado** para fazer o **empréstimo**.
- O sistema deve verificar se o aluno tem **pendências**.
- A **renovação** só é permitida se o livro não estiver **reservado por outro usuário**.
- Um **e-mail de confirmação** é enviado ao final do processo.

Objetivo: Modelar um **Diagrama de Sequência UML** representando o fluxo de **empréstimo e renovação** de cópias de livros, com as mensagens trocadas entre atores e objetos.

Empréstimo e Renovação de Livro

Atores e Objetos Envolvidos

Tipo	Nome	Função
Ator	Aluno	Inicia o processo de empréstimo e renovação.
Objeto	InterfaceUsuario	Tela do sistema usada pelo aluno.
Objeto	SistemaLogin	Verifica se o aluno está autenticado.
Objeto	ControladorEmprestimo	Lógica do empréstimo e renovação.
Objeto	BaseDeDados	Consulta e atualiza status de cópias e alunos.
Objeto	EmailService	Envia confirmação de empréstimo ou renovação.

Empréstimo e Renovação de Livro

Fluxo Principal

- Aluno realiza login
- Solicita o empréstimo de uma cópia
- O sistema verifica pendências
- Confirma o empréstimo
- Permite a renovação se possível
- Envia confirmação por e-mail